

One Henkin Quantifier in the empty vocabulary suffices for undecidability

Konrad Zdanowski,

Faculty of Mathematics and Natural Sciences, School of Exact Sciences,
Cardinal Stefan Wyszyński University in Warsaw

December 22, 2016

Abstract

We prove that there are single Henkin quantifiers such that first order logic augmented by one of these quantifiers is undecidable in the empty vocabulary. Examples of such quantifiers are given.

1 Introduction

In first order logic an existential variable y depends on all universal variables x such that y lies in the scope of x . It follows that we can not express that in a predicate $P(x, y, z, w)$ a variable y depends only on x and w depends only on z . To overcome this restriction Henkin proposed to use quantifiers prefixes in which the ordering of variables is only partial, not linear. Then, we could express dependences as above with the following prefix:

$$\forall x \exists y \forall z \exists w P(x, y, z, w).$$

Henkin, or branched, quantifiers are a way of introducing dependences between variables which are not expressible in first order logic. They occurred to be an interesting extension of first order logic which do not introduce the full power of second order quantification. Henkin quantifiers were examined in various contexts. Jaako Hintikka consider the following sentence of natural language:

“Some relative of each villager and some relative of each townsman hate each other.”

His claim, known as Hintikka’s Thesis, states that the logical form of the sentences as above essentially requires branched quantification. We refer to Gierasimczuk and Szymanik [4] for a recent discussion of Hintikka’s Thesis. In complexity theory branched quantifiers were examined as a way of capturing complexity classes by logics, see Blass and Gurevich [1] and Kołodziejczyk [7].

In this paper we prove that there are single Henkin quantifiers H which give undecidable extension of first order logic already in the empty vocabulary. Previous results by Krynicki and Mostowski and by Mostowski and Zdanowski showed this property only for infinite classes of Henkin quantifiers.

2 Basic notions

We investigate different logics with Henkin quantifiers. The simplest Henkin quantifier has the form

$$\begin{array}{c} \forall x \exists y \\ \forall z \exists w \end{array} .$$

Intuitively, it expresses that the choice of y does not depend on the variable z and the choice of w does not depend on x . More formally we can describe the Henkin prefix as an ordered triple $Q = (A, E, D)$, where A and E are disjoint sets of universal and existential variables, respectively, and $D \subseteq A \times E$ is a dependency relation. We say that a variable $y \in E$ depends on a variable $x \in A$ if $(x, y) \in D$. Further on, we will make no differences between quantifiers and quantifier prefixes.

Example.

$$\begin{array}{c} \forall x \exists y \\ \forall z \exists w \end{array} = (\{x, z\}, \{y, w\}, \{(x, y), (z, w)\}).$$

We denote the above quantifier by H .

The inductive step in the definition of semantics for logic with Henkin quantifiers is as follows. Let $Q = (\{x_1, \dots, x_n\}, \{y_1, \dots, y_k\}, D)$. Then,

$$M \models Q\varphi(x_1, \dots, x_n, y_1, \dots, y_k)$$

if and only if

there are operations f_1, \dots, f_k on M such that

$$(M, \{f_i\}_{i \leq k}) \models \forall \bar{x} \varphi(\bar{x}, f_1(\bar{x}_1), \dots, f_k(\bar{x}_k)),$$

where \bar{x} are all universal variables in Q and \bar{x}_i are variables on which y_i depends in Q .

By \mathcal{H} we denote the family of all Henkin quantifiers. For a family of Henkin quantifiers \mathcal{Q} , $L(\mathcal{Q})$ is an extensions of the first order logic by quantifiers in \mathcal{Q} . For a single quantifier Q we write $L(Q)$ for $L(\{Q\})$.

The logic with Henkin quantifiers was shown to be a strengthening of first order logic by Ehrenfeucht. He showed that one can define the finiteness of the universe by the following sentence.

$$\neg \exists t \begin{array}{l} \forall x \exists y \\ \forall z \exists w \end{array} (y = w \equiv x = z) \wedge (t \neq y).$$

The sentence above is equivalent to the second order sentence

$$\neg \exists t \exists f \forall x, y (f(x) = f(y) \Rightarrow x = y) \wedge \forall x (t \neq f(x))$$

which states that there is no injection of the universe of a given model into itself which is not a bijection.

We have the following theorem relating the semantical power of logic with Henkin quantifiers with that of second order logic. The first dependence was independently proved by Enderton and Walkoe, the second is due to Enderton.

Theorem 1 (see [3], [15]) $\Sigma_1^1 \leq L(\mathcal{H}) \leq \Delta_2^1$, where \mathcal{H} is the family of all Henkin quantifiers.

It should be added that all the inequalities above are strict. The first one is obvious since $L(\mathcal{H})$ is closed on the negation and Σ_1^1 is not. The second one was proven by M. Mostowski in [12] by means of truth definitions. For a simpler argument which works for the empty vocabulary see [14].

We will consider the following kinds of Henkin quantifiers. By $H_n x_1 \dots x_n y_1 \dots y_n$ we denote the quantifier

$$\begin{array}{l} \forall x_1 \exists y_1 \\ \forall x_2 \exists y_2 \\ \dots \dots \dots \\ \forall x_n \exists y_n \end{array}$$

By $E_n x_1 x_2 y_1 \dots y_n z_1 \dots z_n$ we denote

$$\begin{aligned} & \forall x_1 \exists y_1 \dots y_n \\ & \forall x_2 \exists z_1 \dots z_n \end{aligned}$$

By H_ω we denote the family of Henkin quantifiers $\{H_n\}_{n=2,3,\dots}$ and similarly for E_ω .

Clearly, each quantifier E_n can be defined in the logic with quantifier H_n . However, it is not known if for each k there is n such that $L(H_k) \leq L(E_n)$.

Now, we present known results on decidability of different logics with Henkin quantifiers. Our aim is to outline for these logics the boundary between decidable and undecidable.

Theorem 2 ([8]) *Let σ be a monadic vocabulary. Logic $L_\sigma(H_2)$ is decidable.*

Theorem 3 ([8]) *Let σ contains one unary function symbol. Then logic $L_\sigma(H_2)$ is undecidable.*

Theorem 4 ([9]) *Let σ be an infinite monadic vocabulary. Then $L_\sigma(H_4)$ is undecidable.*

The proof of theorem 3 gives an up-to-isomorphism a characterization of the standard model of arithmetic in the language of $L_\sigma(H_2)$. An unary function symbol is intended there to be a successor function. Similarly, definitions of addition and multiplication by means of a successor function are given. In [13] it was observed that also for some finite monadic vocabulary τ one obtain undecidable logic $L_\tau(H_4)$.

As far as the empty vocabulary is concern it was not known whether there exists a single Henkin quantifier Q such that $L_\emptyset(Q)$ is undecidable. The only undecidability results were established for the infinite families H_ω ([9]) and E_ω ([13]).

Theorem 5 ([9],[13]) *Logics $L_\emptyset(H_\omega)$ and $L_\emptyset(E_\omega)$ are undecidable.*

In the next section we prove that there is one Henkin quantifier for which we obtain undecidable logic in the empty vocabulary. We present also examples of such quantifiers.

3 Undecidable logics with one Henkin quantifier

Firstly, we prove that there is a single Henkin quantifier such that the logic with this quantifier is undecidable in the empty vocabulary. Next, we give an estimation of a size of such quantifier. Our proof is a modification of proofs of Theorem 5 as presented in [9] and [13]. Krynicky and Mostowski gave in [9] a reduction of the word problem for semigroups to the tautology problem for $L_\emptyset(\mathbf{H}_\omega)$. We carry out this method in a way which allows us to obtain a single Henkin quantifier \mathbf{H}_n or \mathbf{E}_n such that the logic with this quantifier is undecidable in the empty vocabulary.

Theorem 6 *There is n such that logics $L_\emptyset(\mathbf{H}_n)$ and $L_\emptyset(\mathbf{E}_n)$ are undecidable.*

Proof. Let $\Sigma = \{a, b\}$ be an alphabet and let $E = \{v_i = w_i : i \leq m \wedge w_i, v_i \in \Sigma^*\}$ be a semigroup. The word problem for E is the set of equations $v = w$ of words from Σ^* such that any semigroup satisfying E satisfies also $v = w$. We denote this by $E \models v = w$. Let us fix such a semigroup E that its word problem is undecidable.

For each letter x in Σ we fix a function symbol f_x and by $f \circ g$ we denote the composition of f and g . For a word $c_1 \dots c_k \in \Sigma^*$ we define the translation tr as follows, $tr(c_1 \dots c_k) = f_{c_1} \circ \dots \circ f_{c_k}$.

By the representation theorem for semigroups each semigroup is isomorphic to a semigroup of unary functions with the composition as the semigroup operation. Thus we have that

$$\begin{aligned} E \not\models v = w & \text{ if and only if} \\ \exists M \exists f_a f_b & \text{ unary operations on } M \text{ such that} \\ (M, f_a, f_b) \models & \bigwedge_{i \leq m} \forall x \, tr(v_i)(x) = tr(w_i)(x) \wedge \exists x \, tr(v)(x) \neq tr(w)(x). \end{aligned}$$

Let $v = v_1 \dots v_m$ and $w = w_1 \dots w_k$ be arbitrary words over Σ . Then we can express $\exists f_a \exists f_b \forall x (tr(v)(x) = tr(w)(x))$ by means of some Henkin quantifier \mathbf{H}_n and the following formula

$$\begin{array}{ll} \forall x_1 & \exists y_1 \\ \dots & \dots \\ \forall z_m & \exists y_m \\ \forall z_1 & \exists r_1 \end{array} (\varphi_0 \wedge \varphi_{v=w}),$$

$$\begin{array}{ll} \dots & \dots \\ \forall z_k & \exists r_k \end{array}$$

where

$$\begin{aligned}\varphi_0 &= \bigwedge_{v_i=v_j} (x_i = x_j \Rightarrow y_i = y_j) \wedge \bigwedge_{w_i=w_j} (z_i = z_j \Rightarrow r_i = r_j) \wedge \\ &\quad \bigwedge_{v_i=w_j} (x_i = z_j \Rightarrow y_i = r_j), \\ \varphi_{v=w} &= ((\bigwedge_{1 \leq i < m} (x_i = y_{i+1}) \wedge \bigwedge_{1 \leq i < k} (z_i = r_{i+1})) \Rightarrow (x_m = z_k \Rightarrow y_1 = r_1)).\end{aligned}$$

The formula φ_0 says that the choice functions are the same if their rows represent the same letter. The formula $\varphi_{v=w}$ expresses the fact that if the values of x 's and z 's satisfy the dependences of the diagram below and $x_m = z_k$, then $f_{v_1}(x_1) = f_{w_1}(z_1)$. We may depict it as follows. An arrow of the form $y \xrightarrow{f} z$ indicates that $z = f(y)$. Thus, the predecessor of $\varphi_{v=w}$ expresses the following dependences:

$$\begin{array}{ccccccc} x_m & \xrightarrow{f_{v_m}} & x_{m-1} & \xrightarrow{f_{v_{m-1}}} & \dots & \xrightarrow{f_{v_2}} & x_1 \xrightarrow{f_{v_1}} y_1, \\ z_k & \xrightarrow{f_{w_k}} & z_{k-1} & \xrightarrow{f_{w_{k-1}}} & \dots & \xrightarrow{f_{w_2}} & z_1 \xrightarrow{f_{w_1}} r_1 \end{array}$$

Then, equality $y_1 = r_1$ means that $tr(v)(x_m) = tr(w)(z_k)$. Since x_m and z_k are quantified universally and we assume their equality this is equivalent to $\forall x(tr(v)(x) = tr(w)(x))$.

Next, we choose n big enough to express $\exists f_a f_b (\bigwedge_{i \leq m} \varphi_{v_i=w_i})$ in $L(H_n)$. Now, we need to observe that in order to express $\exists x tr(v)(x) \neq tr(w)(x)$ it suffices to add only first order quantification, no matter how long are words v and w . This is the place when we modify previous constructions in order to stay with a fixed Henkin quantifier. To show this let us assume that the choice functions for y and r below are respectively f_a, f_b and that $v = v_1 \dots v_l$ and $w = w_1 \dots w_k$.

Let us consider the following formula,

$$\begin{array}{c} \forall x \exists y \\ \forall z \exists r \\ \exists t_0 \dots t_l s_0 \dots s_k \forall x_3 \exists y_3 (\varphi_0 \wedge (\bigwedge_{i \leq m} \varphi_{v_i=w_i}) \wedge \varphi_{v \neq w}), \\ \dots \dots \dots \\ \forall x_n \exists y_n \end{array} \quad (1)$$

where

$$\begin{aligned} \varphi_{v \neq w} = & \bigwedge_{v_i=a} (x = t_i \Rightarrow y = t_{i-1}) \wedge \bigwedge_{v_i=b} (z = t_i \Rightarrow r = t_{i-1}) \wedge \\ & \bigwedge_{w_i=a} (x = s_i \Rightarrow y = s_{i-1}) \wedge \bigwedge_{w_i=b} (z = s_i \Rightarrow r = s_{i-1}) \wedge \\ & (t_l = s_k) \wedge (t_0 \neq s_0). \end{aligned}$$

Here, $\varphi_{v \neq w}$ states that we can find in a given semigroup two sequences of elements, t_l, \dots, t_0 and s_k, \dots, s_0 such that the values of terms $tr(v)$ and $tr(w)$ on the t_l and s_k are different. But since $t_l = s_k$, it follows that $\exists x \, tr(v)(x) \neq tr(w)(x)$.

Below we present the dependencies which satisfy the elements of these two sequences as it is described by $\varphi_{v \neq w}$.

$$\begin{aligned} t_m & \xrightarrow{f_{v_m}} t_{m-1} \xrightarrow{f_{v_{m-1}}} \dots \xrightarrow{f_{v_1}} t_0, \\ s_k & \xrightarrow{f_{w_k}} s_{k-1} \xrightarrow{f_{w_{k-1}}} \dots \xrightarrow{f_{w_1}} s_0. \end{aligned}$$

It follows that the formula (1) is satisfiable if and only if there is a semigroup M with generators a, b such that it satisfies all equations from E and $M \models v \neq w$. Therefore, we reduced the problem whether $E \not\models v = w$ to the satisfiability problem for $L(\mathbf{H}_n)$. It should be noted that a similar construction works also in a case of sufficiently large quantifier \mathbf{E}_n . See [13] and below where we construct explicit formulas describing the equations from a given semigroup in the logic $L(\mathbf{E}_n)$. \square

4 An estimation of a size of quantifiers H with undecidable logic $L_\emptyset(H)$

Now, we give an estimation of the value of n for which we get undecidable logics $L(\mathbf{H}_n)$ and $L(\mathbf{E}_n)$. Let C be the semigroup with generators a, b, c, d, e , defined by the following equations:

$$ac = ca, ad = da, bc = cb, bd = db, eca = ce, edb = de, cca = ccae.$$

Ceitin proved that the word problem the semigroup C is undecidable, see [2] or chapter A.4 of [10].

Theorem 7 (Ceitin) *The word problem for C is undecidable.*

Having fixed a single semigroup with undecidable word problem we can explicitly construct a quantifier. Below we describe the formulas with quantifiers H_{12} and E_{10} which express that the functions f_a, \dots, f_e satisfy the equations from the Ceitin's semigroup. It follows that

Theorem 8 *The logics $L_\emptyset(H_{12})$ and $L_\emptyset(E_{10})$ are undecidable.*

Proof. The following formula describes the equations from the semigroup C .

$$\begin{aligned}
& \forall x_a \exists y_a \\
& \forall x'_a \exists y'_a \\
& \forall x_b \exists y_b \\
& \forall x'_b \exists y'_b \\
& \forall x_c \exists y_c \\
& \forall x'_c \exists y'_c \\
& \forall x_d \exists y_d \\
& \forall x'_d \exists y'_d \\
& \forall x_e \exists y_e \\
& \forall x'_e \exists y'_e \\
& \forall x_{cc} \exists y_{cc} \\
& \forall x'_{cc} \exists y'_{cc}
\end{aligned}
(\psi \wedge \varphi \wedge \bigwedge_{i < 7} \varphi_i),$$

where

$$\begin{aligned}
\psi &= \bigwedge_{q \in \{a, b, c, d, e, cc\}} (x_q = x'_q \Rightarrow y_q = y'_q), \\
\varphi &= (x_c = x_{cc} \wedge y_c = x'_c \Rightarrow y'_c = y_{cc}), \\
\varphi_0 &= (x_a = x_c \wedge x'_a = y_c \wedge x'_c = y_a \Rightarrow y'_c = y'_a), \\
\varphi_1 &= (x_a = x_d \wedge x'_a = y_d \wedge x'_d = y_a \Rightarrow y'_d = y'_a), \\
\varphi_2 &= (x_b = x_c \wedge x'_b = y_c \wedge x'_c = y_b \Rightarrow y'_c = y'_b), \\
\varphi_3 &= (x_b = x_d \wedge x'_b = y_d \wedge x'_d = y_b \Rightarrow y'_d = y'_b), \\
\varphi_4 &= (x_a = x'_e \wedge y_a = x_c \wedge y'_e = x'_c \wedge x_e = y_c \Rightarrow y_e = y'_c), \\
\varphi_5 &= (x_b = x'_e \wedge y_b = x_d \wedge y_d = x_e \wedge y'_e = x'_d \Rightarrow y_e = y'_d), \\
\varphi_6 &= (x_a = x'_e \wedge y_a = x_{cc} \wedge y'_e = x'_a \wedge y'_a = x'_{cc} \Rightarrow y_{cc} = y'_{cc}).
\end{aligned}$$

The formula ψ expresses the fact that variables y_q and y'_q describe the same functional dependency, for $q \in \{a, b, c, d, e, cc\}$. The formula φ expresses that the choice function for y_{cc} (and, implicitly, for y'_{cc}) is just a composition of a function for y_c with itself. The formulas φ_i describe the i -th equations from the semigroup C given above. It should be clear that indices of variables indicate what kind of function or a composition of functions they represent.

Now let us describe the equations from C with the quantifier \mathbf{E}_{10} . The formula has the form

$$\begin{array}{l} \forall x_1 \exists y_a \exists y_{ca} \exists y_{da} \exists y_b \exists y_{cb} \exists y_{db} \exists y_e \exists y_{eca} \exists y_{de} \exists y_{cca} \\ \forall x_2 \exists y_c \exists y_{ac} \exists y_d \exists y_{ad} \exists y_{bc} \exists y_{bd} \exists y'_e \exists y'_{cca} \end{array} (\gamma \wedge \gamma_{0123} \wedge \bigwedge_{4 \leq i < 7} \gamma_i).$$

Above, the formula γ establishes that existential variables describe the function compositions according to their subscripts. It has the following form:

$$\begin{aligned} \gamma = & (y_a = x_2 \Rightarrow y_c = y_{ca}) \wedge (y_c = x_1 \Rightarrow y_a = y_{ac}) \wedge (y_a = x_2 \Rightarrow y_{da} = y_d) \wedge \\ & (y_d = x_1 \Rightarrow y_{ad} = y_a) \wedge (y_b = x_2 \Rightarrow y_{cb} = y_c) \wedge (y_c = x_1 \Rightarrow y_b = y_{bc}) \wedge \\ & (y_b = x_2 \Rightarrow y_{db} = y_d) \wedge (y_d = x_1 \Rightarrow y_{bd} = y_b) \wedge (x_1 = x_2 \Rightarrow y_e = y'_e) \\ & (y_{ca} = x_2 \Rightarrow y_{eca} = y'_e) \wedge (y_e = x_2 \Rightarrow y_{de} = y_d) \wedge \\ & (y_{ca} = x_2 \Rightarrow y_{cca} = y_c) \wedge (x_1 = x_2 \Rightarrow y_{cca} = y'_{cca}). \end{aligned}$$

The formulas γ_x , for $x \in \{0123, 4, 5, 6\}$ state that axioms of Ceitin's semigroup C are true for these functions. For brevity we grouped the first four equations into one axiom.

$$\begin{aligned} \gamma_{0123} &= (x_1 = x_2 \Rightarrow (y_{ca} = y_{ac} \wedge y_{ad} = y_{da} \wedge y_{bc} = y_{cb} \wedge y_{db} = y_{bd})), \\ \gamma_4 &= (y_e = x_2 \Rightarrow y_{eca} = y_c), \\ \gamma_5 &= (y_{db} = x_2 \Rightarrow y_{de} = y'_e), \\ \gamma_6 &= (y_e = x_2 \Rightarrow y_{cca} = y'_{cca}). \end{aligned}$$

Now, to express for arbitrary words v, w over the alphabet $\{a, \dots, e\}$ that $C \not\models v = w$ it suffices to follow the proof of theorem 6. One need only to add a proper first order prefix to formulas above and the formula $\varphi_{v \neq w}$. Thus, we reduced the problem whether $C \not\models v = w$ to the satisfiability problem for $L(\mathbf{H}_{12})$ or $L(\mathbf{E}_{10})$. \square

5 Conclusions

We showed that there are single, relatively simple, Henkin quantifiers H such that the first order logic augmented with H is undecidable already in the empty vocabulary. However, there is a considerable gap between the decidable logic $L_\emptyset(\mathbf{H}_2)$ (see [8]) and undecidable logics $L_\emptyset(\mathbf{H}_{12})$ and $L_\emptyset(\mathbf{E}_{10})$. It would be desirable to close this gap or, at least, make it smaller.

Moreover, we did not touch a question of decidability of these logics in finite models. Articles by Gurevich [5] and by Gurevich and Lewis [6] could be a good starting point for investigating this problem in finite models. However, if one aims at small quantifiers it may be better to construct by hand a semigroup with the undecidable word problem in the class of finite semigroups.

Finally, let us mention that Mostowski and Zdanowski proved in [13] that logics $L_\emptyset^k(Q)$, for all k and Q , are decidable in the class of infinite models only. However, we also know that for sufficiently large k and Q no algorithm can be proved in ZFC as deciding the tautology problem for the logic $L_\emptyset^k(Q)$ (see [11]). Here again, the complexity of logics $L_\emptyset^k(Q)$ in finite models is unknown.

References

- [1] A. BLASS and Y. GUREVICH, *Henkin quantifiers and complete problems* in **Annals of Pure and Applied Logic**, 32(1986), pp. 1-16.
- [2] G. CEITIN, *An associative calculus with an insoluble problem of equivalence* in **Trudy Math. Inst. Steklov** 52(1958), pp. 172-189.
- [3] H. B. ENDERTON, *Finite Partially-Ordered Quantifiers*, in **Zeitschrift für Mathematische Logik und Grundlagen der Mathematik** 16(1970), pp. 393-397.
- [4] N. GIERASIMCZUK and J. SZYMANIK, *Branching Quantification v. Two-way Quantification* in **Journal of Semantics**, 26(2009), pp. 367-392.
- [5] Y. GUREVICH, *The word problem for certain classes of semigroups*, in **Algebra and Logic** 5 (1966), pp. 25-35.

- [6] Y. GUREVICH and H. R. LEWIS, *The word problem for cancellation semigroups with zero*, in **Journal of Symbolic Logic** 49 (1984), pp. 184–191.
- [7] L. A. KOŁODZIEJCZYK, *The expressive power of Henkin quantifiers with dualization*, master’s thesis, Institute of Philosophy, Warsaw University, 2002.
- [8] M. KRYNICKI and A. H. LACHLAN, *On the semantics of the Henkin quantifier*, in **Journal of Symbolic Logic** 44 (1979), pp. 184–200.
- [9] M. KRYNICKI and M. MOSTOWSKI, *Decidability problems in language with Henkin quantifiers*, in **Annals of Pure and Applied Logic** 58 (1992), pp. 149–172.
- [10] A. V. MIKHALEV and G. F. PILZ, Eds., *The concise handbook of algebra*, Springer, 2002
- [11] M. MOSTOWSKI, *Pure logic with branched quantifiers*, in **Zeitschrift für Mathematische Logik und Grundlagen der Mathematik** 35 (1989), pp. 45–48.
- [12] M. MOSTOWSKI, *Arithmetic with the Henkin quantifier and its generalizations*, in F. Gaillard, D. Richard, editors, **Seminaire du Laboratoire Logique, Algorithmique et Informatique** Volume II, 1989–1990, pp. 1–25.
- [13] M. MOSTOWSKI and K. ZDANOWSKI, *Degrees of Logics with Henkin Quantifiers in poor Vocabularies*, in **Archive for Mathematical Logic**, 43(2004), pp. 691–702.
- [14] M. MOSTOWSKI and K. ZDANOWSKI, *Henkin Quantifiers in Finite Models*, in preparation.
- [15] W. J. WALKOE, *Finite partially-ordered quantification*, in **Journal of Symbolic Logic** 35(1970), pp. 535–555.